Министерство образования и науки Российской Федерации.

Федеральное государственное бюджетное образовательное учреждение высшего образования.

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ИРКУТСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ.

ОТЧЕТ

к лабораторной работе по дисциплине:

|  |
| --- |
| **Объектно-ориентированное программирование** |
| Разработка классов и их использование |

наименование темы

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Выполнил студент группы: | ИСТб-16-1 |  |  |  | Винзовский М.Е. |
|  | шифр группы | подпись | Фамилия ИО |
| Проверил: | доцент |  | Аршинский В.Л. |
|  | должность | подпись | Фамилия ИО |

**г.Иркутск2018 г.**

1. **Постановка задачи**

Провести объектную декомпозицию задачи из индивидуального задания и разработать класс, содержащий соответствующие поля для хранения необходимых данных и методы, обеспечивающие достаточную для решения задачи функциональность класса. Создание объекта класса решающего задание и вызов его методов должны осуществляться из главного класса. Для передачи исходных данных в экземпляр класса решающего задачу должны быть разработаны соответствующие методы:

a. Найдите символ в строке, который встречается наибольшее число раз.

b. Найдите самое длинное слово в предложении.

c. Удалите в предложении каждое третье слово.

Индивидуальное задание выполняется на основе заданий к лабораторной работе №1. Т.е. конечная программа должна в итоге представлять собой набор классов и каждая подзадача должна быть выделена в отдельный метод.

1. **Описание структур классов**

Класс Lab2- главный класс программы. Осуществляет связь с классом Chars

Класс Chisla предназначен для обработки строковых данных. Используется в 1-ой задаче.

Таблица 1 – поля класса Chisla

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Имя переменной | Назначение  переменной | Тип переменной | ОДЗ |
| 1 | mas | Исходный массив чисел | int | [-2147483648…2147483647] |
| 2 | count | Количество элементов | int | [0…2147483647] |

1. **Описание методов классов и спецификация локальных переменных**

Таблица 2 – таблица спецификаций переменных класса Chars

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Имя** | | **Назначение** | | | | **Тип** | | | **ОДЗ** |
| Chars() – пустой конструктор | | | | | | | | | |
| Chars(char[] chars) – конструктор с вводом массива символов | | | | | | | | | |
| Chars(int length) – конструктор с вводом кол-ва символов и заполнение его пробелами | | | | | | | | | |
| i | Счетчик | | | | | int | | | [0…2147483647] |
| Chars(String string) – конструктор с вводом строки | | | | | | | | | |
| i | Счетчик | | | | | int | | | [0…2147483647] |
| Chars(Chars clone) – конструктор копирования | | | | | | | | | |
| Chars(int n, char ch) – конструктор с вводом кол-ва символов и заполнение его определенном символом | | | | | | | | | |
| print() - метод для вывода строки | | | | | | | | | |
| i | | | Счетчик | | int | | | | [0…2147483647] |
| text | | | Строка для вывода | | String | | | | - |
| maxChar() –метод для нахождения кол-ва символов встречаемое наибольшее число раз | | | | | | | | | |
| i,j | | | | Счетчики | | int | | [0…2147483647] | |
| count | | | | Текущее кол-во одинаковых символов | | int | | [0…2147483647] | |
| max | | | | Максимальное кол-во одинаковых символов | | int | | [0…2147483647] | |
| printCountChar(int count) –метод для вывода символов встречаемые определенное число раз | | | | | | | | | |
| i,j | Счетчики | | | | | | int | | [0…2147483647] |
| c | Текущее кол-во одинаковыхсимволов | | | | | | int | | [0…2147483647] |
| count | Нужное число символов | | | | | | int | | [0…2147483647] |
| text | Строка с символами | | | | | | String | | - |
| maxWord() - метод для нахождения числа символов в самом длинном слове | | | | | | | | | |
| i,j | Счетчики | | | | | | int | | [0…2147483647] |
| count | Текущая длина слова | | | | | | int | | [0…2147483647] |
| max | Максимальная длина слова | | | | | | int | | [0…2147483647] |
| printCountWord(int count) –метод для вывода слов определенной длины | | | | | | | | | |
| i,j | Счетчики | | | | | | int | | [0…2147483647] |
| c | Текущая длина слова | | | | | | int | | [0…2147483647] |
| count | Нужная длина слова | | | | | | int | | [0…2147483647] |
| text | Строка со словами для вывода | | | | | | String | | - |
| ch | Массив для записи текущего слова | | | | | | Char[] | | - |

Продолжение таблицы 2

|  |  |  |  |
| --- | --- | --- | --- |
| deleteWords(int wordsDelete) - метод для удаления каждого wordsDelete слова | | | |
| wordsDelete | Удаление каждого wordsDelete слова | int | [0…2147483647] |
| text | Строка без лишних слов | String | - |
| ch | Новый массив словдля вывода | Chars | - |
| three | Текущее слово | int | [0…2147483647] |
| i,j | Счетчики | int | [0…2147483647] |

1. **Таблица тестов**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № Теста | Исходные данные | | | Результаты | | |
| Строка 1 | Строка 2 | Строка 3 | 1 | 2 | 3 |
| 1 | Other components round Platform Products out the functionality, macromolecule. | Other components round Platform Products out the functionality, macromolecule. | Other components round Platform Products out the functionality, macromolecule. | Символ(ы):  ' '  'o' | Длинаслова: 13  Слова:  Functionality  macromolecule | Удаленокаждое 3-еслово:  Other components Platform Products the functionality, |

1. **Результаты тестирования**

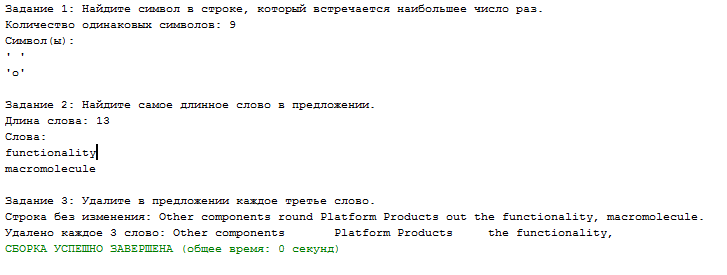


Рисунок 1 – результаты выполнения теста

1. **Исходный код**

Класс Chars:

public class Chars {

privateint length = 0;

private char [] chars = new char[length];

public Chars()

{

this.chars = new char[] {'H','e','l','l','o'};

this.length = this.chars.length;

}

public Chars(char[] chars)

{

this.length = chars.length;

this.chars = chars;

}

public Chars(int length)

{

this.length = length;

this.chars = new char[this.length];

for (inti = 0; i<this.length; i++)

{

this.chars[i]=' ';

}

}

public Chars(String string)

{

this.length = string.length();

this.chars = new char[this.length];

for (inti = 0; i<this.length; i++)

{

this.chars[i] = string.charAt(i);

}

}

public Chars(Chars clone)

{

this.length = clone.length;

this.chars = clone.chars;

}

public Chars(int n, char ch)

{

this.length = n;

this.chars = new char[this.length];

for (inti = 0; i<this.length; i++)

{

this.chars[i] = ch;

}

}

publicintgetLength()

{

return length;

}

public char getChar(inti)

{

returnthis.chars[i];

}

public char[] getChars()

{

returnthis.chars;

}

public void setLength(int length)

{

this.length = length;

}

public void setChar(inti, char ch)

{

this.chars[i] = ch;

}

public void setChars(char[] ch)

{

this.chars = ch;

}

public String print()

{

String text = "";

for (inti=0; i<this.getLength(); i++)

{

text+=this.getChar(i);

}

return text;

}

publicintmaxChar()

{

int max = 0;

int count;

for (inti = 0; i<this.getLength(); i++)

{

count = 0;

for (int j = 0; j<this.getLength(); j++)

{

if (this.getChar(i) == this.getChar(j))

{

count++;

}

}

if (count > max)

{

max = count;

}

}

return max;

}

public String printCountChar(int count)

{

String text = "";

int c;

for (inti=0; i<this.getLength(); i++)

{

c = 0;

for (int j = i; j<this.getLength(); j++)

{

if (this.getChar(i) == this.getChar(j))

{

c++;

}

}

if (c == count)

{

text += "'" + this.getChar(i) + "'\n";

}

}

return text;

}

publicintmaxWord()

{

int max=0;

int count = 0;

for (inti=0; i<this.getLength(); i++)

{

count++;

if (this.getChar(i) == ',' || this.getChar(i) == ':' ||

this.getChar(i) == '.' || this.getChar(i) == ';' || this.getChar(i) == ' ')

{

count--;

if (count > max)

{

max = count;

}

count = 0;

}

if ( i == this.getLength()-1 && count > max)

{

max =count;

}

}

return max;

}

public String printCountWord(int count)

{

String text = "";

int c = 0;

char[] ch = new char[count+1];

for (inti=0; i<this.getLength(); i++)

{

c++;

ch[c-1] = this.getChar(i);

if (this.getChar(i) == ',' || this.getChar(i) == ':' ||

this.getChar(i) == '.' || this.getChar(i) == ';' || this.getChar(i) == ' ')

{

c--;

if (c == count)

{

for (int j=0; j<count; j++)

{

text+=ch[j];

}

text+='\n';

}

c = 0;

}

if ( i == this.getLength()-1 && c == count)

{

for (int j=0; j<count; j++)

{

text+=ch[j];

}

text+='\n';

}

}

return text;

}

public Chars deleteWords(intwordsDelete)

{

String text = "";

int three = 1;

for (inti=0; i<this.getLength(); i++)

{

if (this.getChar(i) == ' ' && three == wordsDelete)

{

three=0;

}

if (this.getChar(i) == ' ' && three != wordsDelete)

{

three++;

}

if (three != wordsDelete)

{

text+=this.getChar(i);

}

}

Chars ch = new Chars(text);

returnch;

}

}

КлассLab2:

public class Lab2 {

public static void main(String [] args)

{

System.out.println("Задание 1: Найдите символ в строке, который встречается наибольшее число раз.");

Chars chars = new Chars("Other components round Platform Products out the functionality, macromolecule. ");

int max = chars.maxChar();

System.out.println("Количествоодинаковыхсимволов: " + max + "\nСимвол(ы):");

System.out.println(chars.printCountChar(max));

System.out.println("Задание 2: Найдите самое длинное слово в предложении.");

max = chars.maxWord();

System.out.println("Длина слова: " + max + "\nСловa:");

System.out.println(chars.printCountWord(max));

System.out.println("Задание 3: Удалите в предложении каждое третье слово.");

intwordDelete = 3;

System.out.print("Строка без изменения: ");

System.out.println(chars.print());

Chars chars2 = chars.deleteWords(wordDelete);

System.out.print("Удаленокаждое " + wordDelete + " слово: ");

System.out.println(chars2.print());

}

}

1. **Список использованной литературы**
2. Объектно-ориентированное программирование. Метод.указания по выполнению лабораторных работ / сост.: В.Л. Аршинский. – Иркутск : Изд-во ИРНИТУ, 2015. – 23 c.
3. Программирование на Java для детей, родителей, дедушек и бабушек/ Файн Я.